

AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1 1. (Currently Amended) A computer system, comprising:
2 a first data storage unit storing a first program and a second program;
3 a second data storage unit storing a product key of the first program according to the second .
4 program, the product key accommodating an installation of the first program, said second data
5 storage unit being separate from said first data storage unit; and
6 a third program stored in the first data storage unit for reinstalling the first program, the third
7 program reading the product key of the first program stored in the second data storage unit, when
8 a product key from the third program and the product key stored in the second data storage unit are
9 identical.

1 2. (Original) The computer system according to claim 1, with the first program being an
2 operating system controlling the operation of the computer system.

1 3. (Original) The computer system according to claim 1, with the first data storage unit
2 comprising:
3 a first unit storing the first program; and

4 a second unit storing the third program.

1 4. (Original) The computer system according to claim 3, with the second program being
2 stored in the first unit or the second unit.

1 5. (Original) The computer system according to claim 3, with the second unit being a re-
2 writable magnetic disk storage device or an optical storage device.

1 6. (Original) The computer system according to claim 1, with information on the product key
2 of the first program being a bar code-readable signal.

1 7. (Original) The computer system according to claim 1, with the third program being
2 provided with an information input window to allow a user to directly input the product key, when
3 a product key from the third program and the product key stored in the second data storage unit being
4 not identical with each other.

1 8. (Original) The computer system according to claim 1, further comprising an extended
2 complementary metal-oxide semiconductor random-access memory, the second data storage unit
3 being accommodated in the extended complementary metal-oxide semiconductor random-access
4 memory.

1 9. (Original) The computer system according to claim 8, with the extended complementary
2 metal-oxide semiconductor random-access memory having an auxiliary power source to preserve
3 the stored information when the computer system is off.

1 10. (Original) The computer system according to claim 1, with the second program being
2 installed in a hard disk drive storing the first program and application programs.

1 11. (Original) The computer system according to claim 1, with the second program being
2 erased when the product key is stored in the second data storage unit to prevent the product key of
3 the first program from being reentered.
a1

1 12. (Original) A method for storing a product key of an operating system program,
2 comprising the steps of:

3 reading the product key comprised of a bar code by a bar code reader, corresponding to an
4 installation process of the operating system program, the operating system program for controlling
5 the operations of a computer system, the computer system comprising a central processing unit, a
6 main memory, a basic input-output system read only memory, an auxiliary memory storing therein
7 information set up by the basic input-output system read only memory, using the operating system
8 program having the product key; and

9 storing the product key in a product key storage by activating a product key storage program.

1 13. (Original) The method according to claim 12, with the product key storage being
2 accommodated in an extended complementary metal-oxide semiconductor random-access memory.

1 14. (Original) The method according to claim 12, with the product key storage program being
2 installed in a hard disk drive storing the operating system program and application programs therein.

1 15. (Original) A method, comprising the steps of:

2 initiating an initial install of a first program on a first data storage unit on a computer system;
3 inputting a product key of the first program, the product key being used for certifying an
4 authenticity of the first program and accommodating an installation of the first program on the
5 computer system;

6 writing the product key onto a second data storage unit of the computer system;

7 installing the remainder of the first program after writing the product key;

8 initiating a reinstallation of the first program on the computer system;

9 reading the product key from the second data storage unit;

10 comparing the product key read from the second data storage unit with the product key of the
11 first program;

12 inputting the product key into a product key input window of the first program when the
13 product keys are compared to be identical; and

14 continuing to complete the reinstallation of the first program after the product key is inputted
15 into the product key input window.

1 16. (Original) The method according to claim 15, with the step of inputting the product key
2 being through a bar code reader from an installation media of the first program.

1 17. (Original) The method according to claim 15, with the step of storing the product key in
2 the second data storage unit being controlled by a second program, the second program being erased
3 after the step of storing the product key to prevent the product key of the first program from being
4 reentered.

1 A1
1 18. (Original) The method according to claim 15, further comprising the step of initiating a
2 checksum of the specific regions of the second data storage unit having the product key to ascertain
3 whether the read product key is correct.

1 19. (Original) The method according to claim 15, with the step of comparing having the
2 product key of the first program obtained from a third program accommodating the reinstallation of
3 the first program.

1 20. (Original) The method according to claim 15, further comprising the step of storing the
2 product key in a specific region of the first data storage unit and the first program continuing to
3 install on the computer system before the step of writing the product key onto a second data storage
4 unit, the product key being written from the product key stored on the first data storage unit.

1 21. (Original) The method according to claim 15, with the first program being an operating
2 system controlling the operation of the computer system.

1 22. (Original) The method according to claim 15, with the step of storing the product key in
2 the first data storage unit being controlled by a second program, the second program being erased
3 after the step of storing the product key to prevent the product key of the first program from being
4 reentered, the step of comparing having the product key of the first program obtained from a third
5 program accommodating the reinstallation of the first program.

1 23. (Original) The method according to claim 22, with the first data storage unit comprising:
2 a first unit storing the first program; and
3 a second unit storing the third program.

1 24. (Original) The method according to claim 23, with the second program being stored
2 in the first unit or the second unit.

1 25. (Original) The method according to claim 23, with the second unit being a re-writable
2 magnetic disk storage device or an optical storage device.

1 26. (Original) The method according to claim 23, with the second program being installed

2 in a hard disk drive storing the first program and application programs.

1 27. (Original) The method according to claim 23, with the third program being provided with
2 the information input window to allow a user to directly input the product key, when the product key
3 from the third program and the product key stored in the second data storage unit being not identical
4 with each other.

1 28. (Original) The method according to claim 15, with the second data storage unit being
2 accommodated in the extended complementary metal-oxide semiconductor random-access memory
3 having a backup power source.
